Guiding Noisy Label Conditional Diffusion Models with Score-based Discriminator Correction

Dat Nguyen Cong
FPT Software AI Center
datnc13@fpt.com

Hieu Tran Bao
FPT IS AI R&D Center
hieutb2@fpt.com

Tung Hoang-Thanh
VNU University of Engineering and Technology
htt210@gmail.com

Abstract

Diffusion models have gained prominence as state-of-the-art techniques for synthesizing images and videos, particularly due to their ability to scale effectively with large datasets. Recent studies have uncovered that these extensive datasets often contain mistakes from manual labeling processes. However, the extent to which such errors compromise the generative capabilities and controllability of diffusion models is not well studied. This paper introduces Score-based Discriminator Correction (SBDC), a guidance technique for aligning noisy pre-trained conditional diffusion models. The guidance is built on discriminator training using adversarial loss, drawing on prior noise detection techniques to assess the authenticity of each sample. We further show that limiting the usage of our guidance to the early phase of the generation process leads to better performance. Our method is computationally efficient, only marginally increases inference time, and does not require retraining diffusion models. Experiments on different noise settings demonstrate the superiority of our method over previous state-of-the-art methods.

1. Introduction

Diffusion models [21, 43] have emerged as state-of-the-art techniques for generating highly realistic and controllable images and videos. Their success is largely attributed to their ability to scale effectively with massive datasets [4, 42, 50]. However, these large and often uncurated datasets come with significant challenges, including mislabeled, ambiguous [6], and unsafe content [48]. As a result, models trained on such data may produce low-quality, irrelevant, or even nonsensical outputs. These data quality issues are prevalent across both labeled datasets like ImageNet [13] and multi-modal datasets such as LAION-5B [42], negatively impacting the performance of generative models. Manually addressing these issues is an extremely costly and nearly infeasible task.

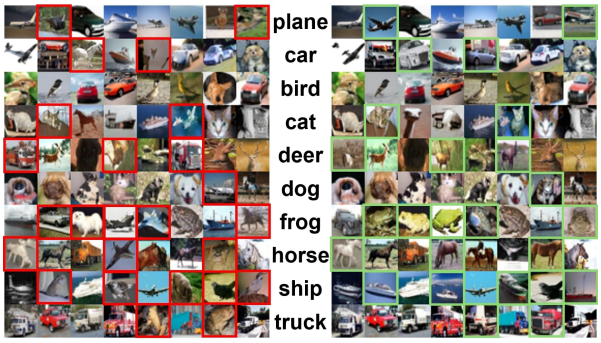


Figure 1. The uncurated generated images of EDM (left) and SBDC (right) on the CIFAR-10 dataset with 50% symmetric noise. We highlight erroneous images their and corrections in red and green squares, respectively.

Noisy datasets can cause conditional generative models to generate lower-quality images that do not align with the conditions (Fig. 1). This paper focuses on methods for correcting the outputs of generative models trained on noisy datasets. Existing approaches often require retraining generative models with modified loss functions (and/or) on modified datasets. Works on noisy conditional generation rely on transition matrix estimation techniques [2, 28, 55, 60], which adjust the objective function to mitigate the effects of noisy labels [22, 34]. This approach involves multi-stage training processes, where errors and biases from one stage can propagate to the next. Alternatively, noise detection methods [11, 63] identify and remove noisy data from the training set and then retrain generative models on the cleaned datasets. Retraining large-scale generative models is extremely expensive and time-consuming, rendering the above methods impractical for large-scale models.

To address this challenge, we propose Score-based Discriminator Correction (SBDC), an inference-time method that enhances the robustness of pre-trained conditional diffusion models against label noise. We train a *small* discriminator and use it to guide the generation of the pre-trained conditional diffusion model (Fig. 1). Our method does not require retraining pre-trained generative models and only incurs a small computational overhead at inference time.

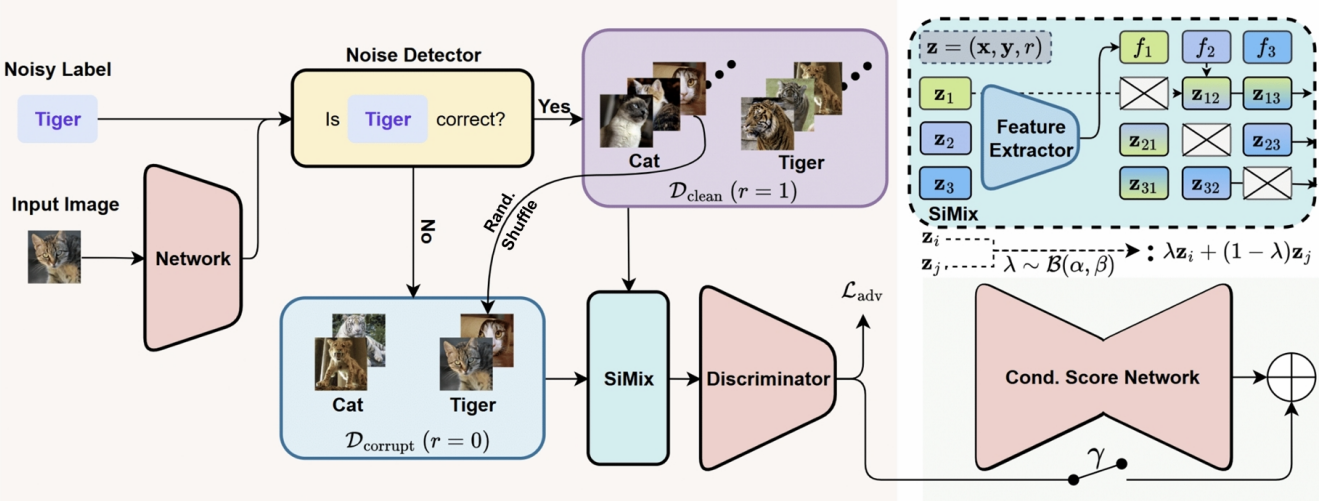


Figure 2. The overall training (left) and inference (bottom right) procedure of the proposed approach. The solid curve from $\mathcal{D}_{\text{clean}}$ to $\mathcal{D}_{\text{corrupt}}$ indicates we randomly shuffle the pseudo-clean label to the corrupted label. In **SiMix**, the nearest neighbor in the feature space is selected to mix with the current input. The discriminator is narrowly applied through a γ-gate during the inference phase.

We show theoretically and empirically that correction guidance can be derived from a discriminator distinguishing correct from incorrect labels.

Our approach is related to Noise-contrastive estimation (NCE) [17], which differentiates real data from fake data for unnormalized models. Taking advantage from it, we utilize prior works on noise detection and streamline the training process. The noise detection process and discriminator training are fast and, thus, are highly suitable for inference time methods. The generation process can be further optimized for inference time efficiency using soft guidance (Sec. 3). Through extensive experiments on multiple datasets—with varying annotation sizes, complexity, and noise levels, we demonstrate that our method consistently outperforms the state-of-the-art Transition-aware Weighted Denoising Score Matching (TDSM) [34] across several metrics, particularly on real-world and diverse-class datasets. We found that TDSM struggles with high-intensity noise, often generating images with incorrect labels. In contrast, our approach effectively corrects these errors, significantly enhancing overall performance (Fig. 7).

Our contributions

- We propose Score-based Discriminator Correction to align pre-trained noisy conditional diffusion models at inference time.
- We empirically demonstrate that class instability usually happens during the early and middle phase of the sampling process, allowing us to apply our method to a fraction of the sampling process.
- Experiments show that SBDC outperforms TDSM, particularly in enhancing intra-class quality, handling high-intensity noise, and is efficient in both post-process training and generation phases.

2. Preliminary

This section provides a concise overview of diffusion models and the problem of learning with noisy labels.

Definition. We denote the train set as $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, where $\mathbf{x}_i \in \mathcal{X}$ is the data instance and $\mathbf{y}_i \in \mathcal{Y}$ is the true label of the i^{th} sample. $\tilde{\mathbf{y}}_i \in \mathcal{Y}$ denotes the observed (and potentially incorrect) label of the i^{th} sample. A conditional generative dataset consists of i.i.d. samples from the joint probability distribution P over $\mathcal{X}\times\mathcal{Y}$. During training, only $\tilde{\mathcal{D}} = \{(\mathbf{x}_i, \tilde{\mathbf{y}}_i)\}_{i=1}^n$ is observed and $\tilde{\mathcal{D}}$ is potentially different from $\mathcal{D}$. We also denote $\mathcal{D}_r = \{(\mathbf{x}, \mathbf{y}_r) \in \tilde{\mathcal{D}} | \mathbf{y}_r = \mathbf{y}\}$ and $p_r \sim \mathcal{D}_r$ as the clean distribution, $\mathcal{D}_f = \{(\mathbf{x}, \mathbf{y}_f) \in \tilde{\mathcal{D}} | \mathbf{y}_f \neq \mathbf{y}\}$ and $p_f \sim \mathcal{D}_f$ as the corrupt distribution.

Diffusion Models. Anderson [1], Song et al. [46] represent the diffusion process through first-order stochastic differential equations. The forward diffusion process in Eq. (1) models the time-dependent displacement $\{\mathbf{x}_t\}_{t=0}^T$

$$d\mathbf{x}_t = \mathrm{f}(\mathbf{x}_t, t)dt + \mathrm{g}(t)d\mathrm{w}_t, \tag{1}$$

where $\mathrm{f}(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d$ is the drift, $\mathrm{g}(\cdot) : \mathbb{R} \to \mathbb{R}$ is the diffusion coefficient, and w_t is a d-dimensional Brownian motion. This process gradually transforms the data distribution $q(\mathbf{x}_0)$ to a simple distribution $q(\mathbf{x}_T)$, typically a Gaussian distribution. The forward process has a unique reverse-time diffusion process which converts $\mathbf{x}_T$ to $\mathbf{x}_0$

$$d\mathbf{x}_t = \left[\mathrm{f}(\mathbf{x}_t, t) - \mathrm{g}^2(t)\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)\right] dt + \mathrm{g}(t)d\bar{\mathrm{w}}_t, \tag{2}$$

where $\bar{\mathrm{w}}_t$ is a d-dimensional reverse-time Brownian motion, $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)$ represents the score of the data instance. Learning the score function $s_\theta(\mathbf{x}_t, t) \approx \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)$ allow sampling from the data distribution through the reverse process. Since it is intractable, Song and Ermon

[45] trains the score network with the per-sample data score $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)$ as the new target, resulting in the denoising score matching (DSM) loss

$$L_{\text{DSM}}(\theta, t) = \mathbb{E}_{p(\mathbf{x}_0)}\mathbb{E}_{p(\mathbf{x}_t|\mathbf{x}_0)}\Big[||\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0) - s_\theta(\mathbf{x}_t, t)||_2^2\Big]. \quad (3)$$

Conditional Diffusion Models with Label Noise. In conditional generation, the target distribution is $p(X|Y)$, where X is generated from the label $Y \in \{1, ..., K\}$. During training, Y is added to the denoising network as auxiliary information to guide the generation process towards the distribution of the target sample. However, in real-world settings, only the noisy label $\tilde{Y}$ is observed. Training the score network with the objective

$$L_{\text{DSM}}(\theta, t) = \mathbb{E}_{p(\mathbf{x}_0,\tilde{\mathbf{y}})}\mathbb{E}_{p(\mathbf{x}_t|\mathbf{x}_0,\tilde{\mathbf{y}})}\Big[||\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0, \tilde{\mathbf{y}}) - s_\theta(\mathbf{x}_t, \tilde{\mathbf{y}}, t)||_2^2\Big], \quad (4)$$

eventually makes $s_\theta(\mathbf{x}_t, \tilde{\mathbf{y}}, t)$, or $s_\theta(\mathbf{x}_t, \tilde{\mathbf{y}})$ in short, learn the noisy distribution $p(X|\tilde{Y})$.

Refining Generative Process in Inference Time. As recent models continue to grow in size, retraining to reduce approximation errors becomes impractical. Discriminator Guidance [24] refines the score function using discriminator signals to distinguish real from synthesized samples

$$\nabla_{\mathbf{x}_t} \log p^\phi_\theta(\mathbf{x}_t) = s_\theta(\mathbf{x}_t) + w \log \frac{D^t_\phi(\mathbf{x}_t)}{1 - D^t_\phi(\mathbf{x}_t)}, \quad (5)$$

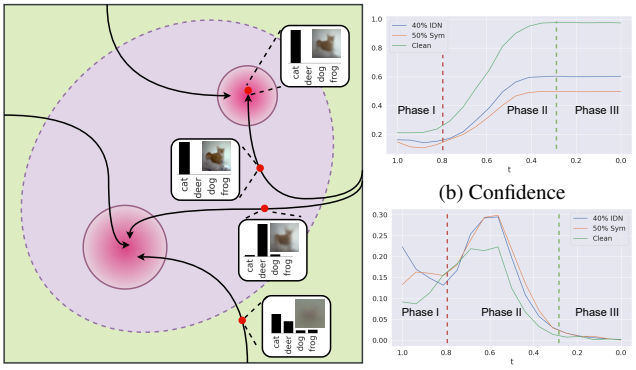
where w is a constant guidance weight. Sampling with the new score reduces the model's approximation error. We also use a discriminator, but train the discriminator solely on real data for direct alignment estimation $\nabla \log \frac{p(\mathbf{x}_t|\mathbf{y})}{p(\mathbf{x}_t|\tilde{\mathbf{y}})}$.

3. Methodologies

In this section, we present our approach to training the discriminator and correcting the score estimation during inference with it. In Sec. 3.1, we study the effect of training bias on the deviation of generation trajectory from the correct path. Sec. 3.2 introduces the discriminator correction to guide the target distribution trajectory. To boost the inference time, we limit the discriminator usage at a specific interval (Sec. 3.3). The practical training implementation is presented in Sec. 3.4. We illustrate the training and inference procedure in Fig. 2.

3.1. Noisy Conditional Generative Behavior

During the generation process, the diffusion model first determines the update direction by estimating the fully denoised sample $\mathbf{x}_\theta(\mathbf{x}_t, \mathbf{y})$ from $\mathbf{x}_t$, and then updates the next state following Eq. (2). The updated



(a) Conditional target field (b) Confidence (c) Instability

Figure 3. (a): Illustration of the three phases in a two-mode distribution. (b,c): Estimated $C(t)$ and $I(t)$. We use 18-step Heun sampling (normalizing the maximum value to 1) and the reverse process of 2k samples for each noise setting.

direction can be viewed as a Monte-Carlo estimator $\nabla_{\mathbf{x}_t} \log p_\theta(\mathbf{x}_t|\mathbf{y}) = \mathbb{E}_{p_\theta(\mathbf{x}_0|\mathbf{x}_t,\mathbf{y})}[\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)] \approx \frac{1}{n}\sum_{i=1}^n \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}^{(i)}_\theta)$. Typically, $\mathbf{x}^{(i)}_\theta$ is sampled from $p_\theta(\cdot|\mathbf{x}_t, \mathbf{y})$ and $n = 1$, meaning the *influence* of a single pair $(\mathbf{x}_0, \mathbf{y})$ on $\mathbf{x}_t$. This results in high bias to the individual target sample during sampling, which is crucially harmful in the mislabeled cases.

We quantitatively characterize the variations of individual targets $\mathbf{x}_\theta(\mathbf{x}_t, \mathbf{y})$ at time t with the *confidence* (C) and the *instability* (I) generation probabilities, defined as

$$C(t) = \mathbb{P}\Big[f(\mathbf{x}_\theta(\mathbf{x}_t, \mathbf{y})) = \mathbf{y}\Big], \quad (6)$$

$$I(t) = \mathbb{P}\Big[f(\mathbf{x}_\theta(\mathbf{x}_t, \mathbf{y})) \neq f(\mathbf{x}_\theta(\mathbf{x}_{t-1}, \mathbf{y}))\Big], \quad (7)$$

where $f(\cdot)$ predicts the class of the denoised sample $\mathbf{x}_\theta$ at step t. In practice, $f(\cdot)$ is often a large pre-trained classifier, since it remains robust to blurry intermediate samples $\mathbf{x}_\theta$ with high confidence (see Fig. 3a).

As shown in Fig. 3a, the sampling process is partitioned into marginalized, conditional, and fine-grained phases (Phase I∼III respectively). Intuitively, $I(t)$ peaks in the conditional phase (Phase II), where multiple modes in the noisy data distribution have comparable influences on the perturbation identity, resulting in unstable targets. In Phase I, t is large and $C(t)$ is low because the labels are mostly discarded from the conditional distribution: $\lim_{t\to\infty} p(\mathbf{x}_t|\mathbf{x}_0, \mathbf{y}) = \lim_{t\to\infty} p(\mathbf{x}_t|\mathbf{x}_0) = p(\mathbf{x}_\infty) \approx p(\mathbf{x}_T)$. In Phase III, the posterior $p(\mathbf{x}_0|\mathbf{x}_t, \mathbf{y})$ concentrates around a single target, thus $C(t)$ solely reflects the noise rate with low $I(t)$. Figs. 3b and 3c empirically validate this argument. They show the estimated $C(t)$ and $I(t)$ for samples generated from models trained on perturbed CIFAR-10 dataset [27]. Essentially, $I(t)$ peaks in Phase II, demon-

strating a large class variation. Since the variation drops as the sampling proceeds, this error remains till the end. Our method for measuring $C(t)$ and $I(t)$ is described in Algorithm 4 (see Appendix C).

In Sec. 3.2, we intervene in the sampling trajectory by using additional auxiliary signals to keep the trajectory on the correct path. From the analysis in Fig. 3, we limit the guidance signal to the conditional phase only and present a simple strategy to do so in Sec. 3.3.

3.2. Score-Based Discriminator Correction

Assuming the score network $s_\theta(\mathbf{x}_t, \tilde{\mathbf{y}})$ perfectly learns the noisy distribution $p(\mathbf{x}_t, \tilde{\mathbf{y}})$, we recover the clean distribution generative process by a simple modification

$$\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{y}) = \nabla_{\mathbf{x}_t} \log p_\theta(\mathbf{x}_t|\tilde{\mathbf{y}}) + \nabla_{\mathbf{x}_t} \log \frac{p(\mathbf{x}_t|\mathbf{y})}{p_\theta(\mathbf{x}_t|\tilde{\mathbf{y}})} \tag{8}$$

where p_θ is the distribution learned from s_θ. Following Kim et al. [24], we approximate the intractable term $\nabla_{\mathbf{x}_t} \log {p(\mathbf{x}_t|\mathbf{y})}/{p_\theta(\mathbf{x}_t|\tilde{\mathbf{y}})}$ with the gradient from a time-dependent discriminator $D^t_\theta(\mathbf{x}_t, \mathbf{y})$, which is trained to discriminate between real and synthetic data.

In our case, we define real as clean data sampled from p_r and fake as corrupted data sampled from p_f. From the definition in Sec. 2, p_r and p_f can be seen as the truncated distribution of $p(\mathbf{x}, \mathbf{y}|\tilde{\mathcal{D}})$. Hence, we formulate the discriminator's training objective as the time-weighted binary cross-entropy loss given clean and corrupt samples:

$$\begin{aligned}\mathcal{L}_{\text{adv}} = &\mathbb{E}_{t,(\mathbf{x},\mathbf{y})\sim p_r,\mathbf{x}_t}\left[-\log D^t_\theta(\mathbf{x}_t, \mathbf{y})\right] \\ &+ \mathbb{E}_{t,(\mathbf{x},\mathbf{y})\sim p_f,\mathbf{x}_t}\left[-\log(1 - D^t_\theta(\mathbf{x}_t, \mathbf{y}))\right].\end{aligned} \tag{9}$$

Following this objective, we use a noise detector to filter clean and corrupt data from the training set and assign each instance in each set the "verified label" r, where $r = 1$ corresponds to the pseudo-clean instance and vice versa. Theorem 1 shows that when the noise rate is not overwhelming, training the discriminator with the objective in Eq. (9) allows the estimation of the log-likelihood ratio.

Theorem 1 *Let $(\cdot, \mathbf{y}_r) \sim p_r$, $(\cdot, \mathbf{y}_f) \sim p_f$, and $D_\theta(\cdot, \cdot) = \sigma(g_\theta(\cdot, \cdot))$ be the logistic function of the discriminator. Assume g satisfies the Lipschitz condition: there exists $L > 0$ such that for all $t \in [\epsilon, T]$, $\mathbf{x}$, $\mathbf{z}$, and $\mathbf{y}$, we have $||g(\mathbf{x}, \mathbf{y}, t) - g(\mathbf{z}, \mathbf{y}, t)||_2 \leq L||\mathbf{x} - \mathbf{z}||_2$. Then, given an optimally trained D_{θ^*}, we have*

$$\begin{aligned}&\mathop{\mathbb{E}}_{\substack{\mathbf{x}_t,\mathbf{y},\tilde{\mathbf{y}},\\ \mathbf{y}_r,\mathbf{y}_f}}\left[\left\|\nabla_{\mathbf{x}_t} \log \frac{D_{\theta^*}(\mathbf{x}_t, \mathbf{y}_r)}{D_{\theta^*}(\mathbf{x}_t, \mathbf{y}_f)} - \nabla_{\mathbf{x}_t} \log \frac{p(\mathbf{x}_t|\mathbf{y})}{p(\mathbf{x}_t|\tilde{\mathbf{y}})}\right\|_2^2\right] \\ &\leq L + \mathop{\mathbb{E}}_{\mathbf{x}_t,\mathbf{y},\tilde{\mathbf{y}}}\left[\left\|\nabla_{\mathbf{x}_t} \log \frac{p(\mathbf{x}_t|\mathbf{y})}{p(\mathbf{x}_t|\tilde{\mathbf{y}})}\right\|_2^2\right]\end{aligned} \tag{10}$$

Algorithm 1 Discriminator Training

Construct $\mathcal{D}_r = \{\mathbf{z}_i = (\mathbf{x}_i, \mathbf{y}_i, 1)\}_{i=1}^M$ and $\mathcal{D}_f = \{\mathbf{z}_j = (\mathbf{x}_j, \mathbf{y}_j, 0)\}_{j=1}^N$ from noise detection method.
while not converged **do**
  Sample $\mathcal{Z} = \{\mathbf{z}^c_i\}_{i=1}^m \cup \{\mathbf{z}^l_i\}_{i=1}^n$ from $\mathcal{D}_r \cup \mathcal{D}_f$
  Random shuffle $\{\mathbf{z}^c_i\}_{i=1}^k \leftarrow \{(\mathbf{x}_i, \tilde{\mathbf{y}}_i, 0)\}_{i=1}^k$ ▷ Pseudo-clean Shuffle
  $\mathcal{Z} \leftarrow \textbf{Simix}(\mathcal{Z})$ ▷ SiMix Augmentation
  Sample $t_1, \ldots, t_{m+n}$ from $[1..T]$
  for $\epsilon_i \sim \mathcal{N}(0, \mathbf{I}), \forall i = 1, \ldots, m+n$ **do**
    $\mathbf{x}_i^{t_i} \leftarrow \sqrt{\bar{\alpha}_{t_i}}\mathbf{x}_i + \sqrt{1 - \bar{\alpha}_{t_i}}\epsilon_i$
  end for
  Calculate $\hat{\mathcal{L}}_\phi \leftarrow -\sum_{i=1}^{m+n} r_i \log D^{t_i}_\phi(\mathbf{x}_i^{t_i}, \mathbf{y}_i) + (1 - r_i)\log(1 - D^{t_i}_\phi(\mathbf{x}_i^{t_i}, \mathbf{y}_i))$
  Update $\phi \leftarrow \phi - \frac{\partial \hat{\mathcal{L}}_\phi}{\partial \phi}$
end while

Algorithm 2 SiMix($\mathcal{Z}$)

$\mathcal{F} = \{f_i \mid f_i = \text{Encode}(\mathbf{x}_i), \forall \mathbf{z}_i := (\mathbf{x}_i, \mathbf{y}_i, r_i) \in \mathcal{Z}\}$
Sample $\lambda_1, \ldots, \lambda_B$ from $\text{Beta}(\alpha, \alpha)$
for i in $[1..B]$ **do**
  $\mathbf{z}_i \leftarrow \lambda_i \mathbf{z}_i + (1 - \lambda_i)\mathbf{z}_{\substack{\arg\min \|f_i - f_j\|_2 \\ j \in [1..B]}}$
end for
Return new $\mathcal{Z}$

From Theorem 1, we ideally assign an auxiliary label r for each instance utilizing previous noise detection methods in classification tasks [25, 36, 38, 63]. The proof is presented in Appendix A.

3.3. Limited Interval Guidance Improves Sample Quality

From the observation in Sec. 3.1, the instability of the generation process is fuzzy in the marginalized phase (high noise levels), extremely low in the fine-grained phase (low noise levels), and peaks in the conditional phase. Hence, we schedule the guidance only in the middle phase.

Kynkäänniemi et al. [29] shows that classifier-free guidance also has a sweet spot for optimal guidance interval. Motivated by their schedule, we formulate the guidance as:

$$\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{y}) = s_\theta(\mathbf{x}_t, \tilde{\mathbf{y}}) + \gamma(t) \log \frac{D^t_\theta(\mathbf{x}_t, \tilde{\mathbf{y}})}{1 - D^t_\theta(\mathbf{x}_t, \tilde{\mathbf{y}})}, \tag{11}$$

$$\gamma(t) = \begin{cases} \gamma & \text{if } t \in (S_{\text{clip\_min}}, S_{\text{clip\_max}}] \\ 0 & \text{otherwise.} \end{cases}$$

The gate $\gamma(\cdot)$ enforces that D is used at $S_{\text{clip\_min}} \leq t \leq S_{\text{clip\_max}}$. This guidance technique is similar to Classifier Guidance proposed by Dhariwal and Nichol [14]. In prac-

CIFAR-10	EDM	TDSM	SBDC	EDM	TDSM	SBDC	EDM	TDSM	SBDC
Metrics	Aymmetric‡			Symmetric			Symmetric		
	20%			50%			80%		
FID	(↓) **1.96** ± 0.02	2.36 ± 0.02	2.49 ± 0.01	**2.07** ± 0.02	2.43 ± 0.03	2.24 ± 0.01	2.15 ± 0.04	**2.25** ± 0.02	2.30 ± 0.21
IS	(↑) 9.95 ± 0.01	10.04 ± 0.08	**10.06** ± 0.03	9.69 ± 0.06	9.84 ± 0.03	**9.87** ± 0.03	9.67 ± 0.06	9.76 ± 0.02	**9.71** ± 0.11
Density	(↑) 103.9 ± 0.40	115.9 ± 0.07	**118.1** ± 0.49	103.6 ± 0.42	**114.0** ± 1.22	112.8 ± 0.48	103.2 ± 0.16	**108.0** ± 0.47	104.1 ± 0.46
Coverage	(↑) 83.7 ± 0.09	**85.1** ± 0.11	84.9 ± 0.04	83.2 ± 0.09	**84.3** ± 0.21	84.0 ± 0.22	82.8 ± 0.13	**83.9** ± 0.03	82.7 ± 0.33
CW-FID	(↓) 11.3 ± 0.03	10.9 ± 0.02	**10.6** ± 0.02	38.6 ± 0.18	18.2 ± 0.16	**15.6** ± 0.10	71.7 ± 0.09	59.8 ± 0.15	**48.2** ± 0.47
CW-Den.	(↑) 98.6 ± 0.42	113.1 ± 0.17	**114.8** ± 0.60	66.8 ± 0.87	95.8 ± 0.97	**98.1** ± 0.34	43.0 ± 0.12	52.0 ± 0.36	**58.0** ± 0.11
CW-Cov.	(↑) 82.3 ± 0.08	**84.4** ± 0.07	83.8 ± 0.02	68.9 ± 0.42	79.5 ± 0.42	**79.9** ± 0.18	44.9 ± 0.71	55.9 ± 0.62	**59.1** ± 0.08
Metrics	Instance			Instance			Average*		
	20%			40%			EDM	TDSM	SBDC	Clean
FID	(↓) **1.95** ± 0.02	2.53 ± 0.01	2.44 ± 0.02	**1.97** ± 0.02	2.20 ± 0.02	2.49 ± 0.01	**2.02**	2.34	2.39	1.88
IS	(↑) 9.80 ± 0.05	9.73 ± 0.01	**9.87** ± 0.03	9.77 ± 0.04	9.85 ± 0.03	**9.88** ± 0.01	9.78	9.84	**9.88**	9.96
Density	(↑) 103.2 ± 0.19	113.1 ± 0.36	**119.7** ± 0.57	102.7 ± 0.53	112.0 ± 0.18	**116.6** ± 0.44	103.3	112.6	**114.2**	104.4
Coverage	(↑) 83.2 ± 0.17	84.0 ± 0.13	**84.4** ± 0.06	83.0 ± 0.19	**84.4** ± 0.10	84.1 ± 0.12	83.2	**84.3**	84.0	83.9
CW-FID	(↓) 16.3 ± 0.18	11.9 ± 0.10	**11.6** ± 0.06	29.7 ± 0.05	18.1 ± 0.04	**13.6** ± 0.15	33.5	23.8	**19.9**	9.8
CW-Den.	(↑) 90.1 ± 0.26	106.5 ± 0.38	**114.9** ± 0.19	74.9 ± 0.50	94.2 ± 0.31	**106.9** ± 0.45	74.7	92.3	**98.5**	103.3
CW-Cov.	(↑) 79.6 ± 0.13	82.4 ± 0.23	**83.1** ± 0.16	73.5 ± 0.22	80.0 ± 0.09	**81.5** ± 0.20	69.8	76.4	**77.5**	83.5

Table 1. Performance comparison on CIFAR-10 dataset under various noise conditions. The percentage below noise type represents the noise rate. The variance is computed through 3 seeds of measurement. The best results are in **bold** and the second best ones are in underlined. *Average results across five different settings. ‡ implies SBDC used CL, while others used CORES.

tice, every noise schedule is discretized into N-step sampling, where $N \leq T$. Hence, the interval is rounded to the closest sampling steps between $S_{\text{clip\_min}}$ and $S_{\text{clip\_max}}$. Our experimental results are conducted on limited guidance, showing the trade-off between quality and inference efficiency. The optimal hyper-parameters are searched through FID evaluation and remain consistent across a wide range. The details are shown in Sec. 4.5.

3.4. Improved Technique for Training Efficiency

Training Pipeline. Figure 2 and Algorithm 1 illustrate the overall training pipeline for the discriminator. Initially, the noisy training data $\tilde{\mathcal{D}}$ are filtered into clean subset $\hat{\mathcal{D}}_r$ and corrupt subset $\hat{\mathcal{D}}_f$ with a noise detector (noise is imperfectly filtered). Then, we augment both training subsets with two proposed data augmentations before feeding it to the discriminator.

Pseudo-clean Shuffle. In practice, unknown and subtle noise rate leads to a high possibility of data imbalance [53]. In a low noise rate setting, we enrich the corrupt subset by simply swapping the label of clean samples and flip r from 1 to 0 accordingly during training. Using this simple augmentation, $|\hat{\mathcal{D}}_f|$ is scaled proportionally to the label space K. Furthermore, this augmentation improves the robustness of the discriminator to unseen noise in the corrupt set. Essentially, *it breaks the instance-dependency of several noise types* (such as Asymmetric noise).

SiMix. Since the discriminator guided the score direction to avoid the corrupt distribution, the diversity of the generation might deteriorate. We propose **SiMix** (Similarity based MixUp), a new inter-class augmentation inspired by MixUp [59], to promote the diversity of corrupt data. Given (instance, label) pairs, we combine instances with high structural similarity in the feature space instead of random selection. SiMix stabilizes the learning process while preventing the discriminator's gradient from exploding, as $\log \mathrm{D}(\cdot)$ goes up significantly when $\mathrm{D}(\cdot) \to 0$. The detail is shown in Algorithm 2.

4. Experiments

4.1. Experiment Setup

To evaluate our method, we conduct experiments with synthetic noise on CIFAR-10, CIFAR-100 [27], and Tiny-Imagenet [30], including Symmetric [26], Asymmetric [18], and Instance-dependent [52] noises at various scales. We synthesize CIFAR-10 noisy data following Bae et al. [2], and use synthesized CIFAR-100 from TDSM [34]. Tiny-Imagenet is perturbed with 20% Symmetric noise, and the results are reported in Appendix F.3. Synthetic noise details are described in Appendix B. We also experiment on human-annotated noise on real-world datasets, including FOOD101 [7] and Clothing-1M [54].

Noise Detectors. We use CORES [11] and CL [36] to explore the effect of different noise detection methods on discriminator performance. The implementation details and results of CORES and CL on each setting are reported in Appendix F.2.

Evaluation metrics. We use Inception Score (IS) [5, 41] and Frechet Inception Distance (FID) [20] to evaluate the

CIFAR-100		EDM (†)	TDSM (*)	SBDC (Ours)	EDM (†)	TDSM (*)	SBDC (Ours)	EDM (†)
Metrics		Symmetric 20%			Symmetric 40%			Clean
FID	(↓)	**2.81**	4.18	3.55	**3.23**	6.84	3.64	2.35
IS	(↑)	12.15	12.19	**12.64**	11.83	11.96	**12.36**	12.80
Density	(↑)	85.24	88.54	**98.91**	85.00	90.05	**95.16**	90.60
Coverage	(↑)	77.71	76.99	**79.42**	76.64	73.92	**78.51**	80.42
CW-FID	(↓)	77.64	76.77	**68.94**	97.96	91.13	**77.13**	64.66
CW-Den.	(↑)	68.20	72.21	**90.48**	51.39	61.27	**76.76**	85.29
CW-Cov.	(↑)	72.13	72.36	**76.61**	63.26	65.50	**72.03**	78.75

Table 2. Performance comparison on CIFAR-100 dataset under various noise conditions. The percentage below noise type represents the noise rate. The best results are in **bold** and the second best ones are in underlined. * indicates that we take the results from their checkpoints.

sampling quality by comparing distributions of generated and real images. To assess fidelity and diversity, we apply Density (Den.) and Coverage (Cov.) [35]. Since class-wise insights are essential, we adopt CW-FID, CW-Density (CW-Den.), and CW-Coverage (CW-Cov.) following [8, 34].

Baselines. We compare our method with the original model (EDM) training with Eq. (5) and the state-of-the-art method (TDSM) to demonstrate the effectiveness of our model. Furthermore, we retrain the score network on filtered clean data and denote it as "Oracle". We also compare retraining relabeled datasets from noise detection methods. GAN-based methods [22, 47] are not reported here because they are prone to mode collapse, and we were unsuccessful in reproducing their experimental results.

Model Configuration. The score model architecture is the same as the one used in EDM [23], while the discriminator architecture is modeled the same as the classifier in TDSM except for the logit layer.

Data Augmentation. For Pseudo-clean Shuffle, based on the noise rate estimated by the noise detector, we apply it to the data with 20% asymmetric noise and 20% IDN noise at a random shuffle rate of 30% to balance the clean and corrupt distribution. For SiMix, λ is sampled from Beta(α, α). In all our experiments, we set α to 0.2 and applied SiMix on half of the mini-batch, while the remaining data remained unchanged. We used CLIP [39] image encoder ViT-B/32 as the feature extractor for Encode$(\cdot)$ during our experiments.

4.2. Results On CIFAR-10 And CIFAR-100

Tables 1 and 2 show the performance of SBDC on CIFAR-10 and CIFAR-100 across various noise settings. The results indicate SBDC outperforms the state-of-the-art method across class-wise metrics while remaining competitive in the remaining metrics shown in Tab. 1. The results exhibit relatively small deviations, ensuring consistency between experiments. Crucially, SBDC offers a strong performance against small noise settings. We show in Sec. 4.5 that the significant improvement mostly contributed to the proposed augmentation methods.

CIFAR-10		Oracle	Ours	Oracle	Ours	Relabel	Oracle	Ours
Metric		Clean 1.5%		Symmetric 20%		Symmetric 40%		
FID	(↓)	**1.88**	3.19	**2.01**	2.54	2.83	**2.17**	2.49
IS	(↑)	9.98	**10.06**	9.91	**10.01**	9.70	9.76	**9.88**
Den.	(↑)	105.24	**124.40**	105.82	**116.38**	105.00	105.54	**116.24**
Cov.	(↑)	84.10	**85.23**	83.76	**83.96**	82.12	83.49	**83.94**

Table 3. Results on the original CIFAR-10, noisy CIFAR-10 with Relabeled training, and noisy CIFAR-10 with Oracle training.

The experiments in Tab. 2 show that our method outperforms the state-of-the-art in all metrics, with FID increasing only 26.33% compared to 48.75% of TDSM for the baseline of 20% noise ratio and 12.69% compared to 111.76% for the baseline of 40% noise ratio. The modest improvement of TDSM is likely due to approximating the objective function with a much smaller number of classes, leading to deviations in calculating the actual objective. This demonstrates that our method provides a better solution for datasets with a large number of classes.

Table 3 shows that our method delivers superior quality and diversity compared to "Oracle" training and relabeling. While "Oracle" training reduces noisy labels, it may discard clean samples, limiting diversity. Relabeling leverages noisy data but depends on the relabeling method. In contrast, our approach directly utilizes noisy data via augmentation while mitigating false generations.

4.3. Results on Original Annotated Data

Figure 4. Ambiguous samples of CIFAR-10, filtered by noise detector. The title above images denote ground-truth labels.

To explore potentially noisy or ambiguous labels in the original datasets, we apply CORES to automatically detect it. From 1.5% of noisy classified samples, we predict the confidence of their label prediction and visualize 16 images with the lowest confidence in Fig. 4. It shows that original data also contains examples with noisy or ambiguous labels. We either apply SBDC or "Oracle" training on them. Table 3 shows improved results, indicating that correcting or filtering noisy data enhances learning.

		EDM	TDSM	SBDC	EDM	TDSM	SBDC
Metrics		CLOTHING1M			FOOD101		
FID	(↓)	4.34	6.75	**4.17**	**3.74**	5.21	4.41
IS	(↑)	3.68	3.66	**3.70**	3.93	**4.39**	4.05
Density	(↑)	98.97	**115.8**	101.3	121.4	113.48	**127.4**
Coverage	(↑)	74.80	74.39	**75.53**	76.80	75.84	**76.94**

Table 4. Performance comparison on CLOTHING1M and FOOD101 datasets. We retrain the baseline and TDSM from scratch with their settings.

Metrics	P	R	F1	Den.	Cov.	CW-Den.	CW-Cov.
	(↑)	(↑)	(↑)	(↑)	(↑)	(↑)	(↑)
CORES.	**87.8**	58.5	**70.2**	107.4	83.7	104.5	82.7
CL	51.1	**83.1**	63.3	**110.5**	**84.0**	**108.3**	**83.4**

Table 5. Performance comparison between CORES and CL without using SiMix on CIFAR-10 with Asymmetric noise.

4.4. Results On Real World Dataset

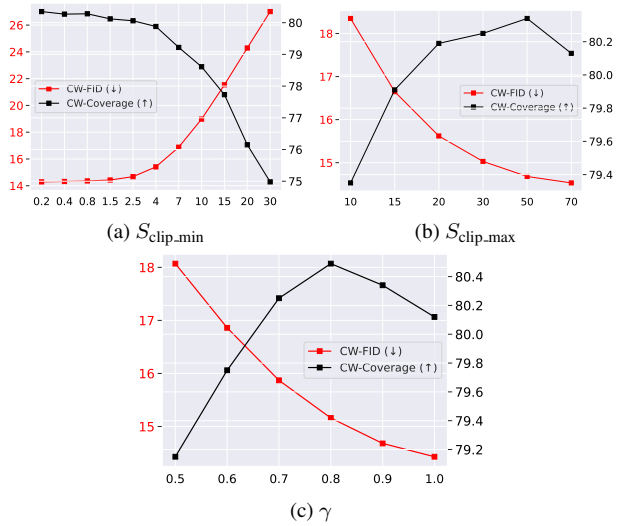
After filtering clean data (see Appendix F.3), we train a ResNet50-based discriminator at 64× resolution for 150 epochs, evaluating every 50 epochs and reporting the best results. Table 4 shows our method outperforms TDSM on most metrics—TDSM struggles on the multi-class FOOD101 dataset, while our approach remains consistently effective, with SBDC yielding superior FID scores.

4.5. Ablation Study

Noise Detection Impact. We compare CORES and CL on asymmetric noise, as different noise prediction methods impact performance. Precision (P) measures accuracy, Recall (R) reflects detected noisy samples, and F1 evaluates overall performance. Table 5 shows that higher Recall improves SBDC, emphasizing the importance of detecting more noisy data over accuracy. This underscores the need for better noise prediction methods.

Limited Interval Guidance. Figures 5a and 5b show the changes in CW-FID and CW-Coverage according to $S_{\text{clip\_min}}$ and $S_{\text{clip\_max}}$. The results indicate that SBDC has a negligible impact even at a large $S_{\text{clip\_min}}$, while $S_{\text{clip\_max}}$ demonstrates a clear sweet-spot for optimal quality. Hence, we set $S_{\text{clip\_min}} = 1.5$ and $S_{\text{clip\_min}} = 50.0$, corresponding to the guidance applied from Step 8 to 16 of the 18 steps for all experiments, including real-world experiments. Furthermore, we discuss the extension of this wise hyper-parameter searches in other pre-trained DMs in Appendix F.6. Some visualization on the intermediate phase of generated samples with/without guidance is shown in Appendix F.10.

Guidance Strength. Figure 5c show the optimal γ search. A high γ value likely leads to a mode drop, similar to Classifier Guidance. Hence, we set $\gamma = 0.9$ for balancing between



(a) $S_{\text{clip\_min}}$ (b) $S_{\text{clip\_max}}$

(c) γ

Figure 5. Ablation studies for several parameters, termination point $S_{\text{clip\_min}}$, starting point $S_{\text{clip\_max}}$, guidance strength γ.

SiMix	Pseudo Shuffle	CW-Den.	CW-Cov.
		102.95	82.70
✓		104.49	82.76
✓	✓	**114.80**	**83.10**

Table 6. Ablation of proposed data augmentation. Results are reported on CIFAR-10 20% Asymmetric noise.

diversity and quality during our all experiments, including real-world experiments.

Data Augmentation. As shown in Tab. 6, SiMix offers better intra-class quality, likely because the artifact is avoided when mixing instances together. We show this through a small example, where mixing two images $(\mathbf{x}_1, \mathbf{y})$ and $(\mathbf{x}_2, \mathbf{y})$ from the same class ($r_1 = 1$, $r_2 = 1$) yields

$$r = \lambda r_1 + (1 - \lambda) r_2 = 1, \tag{12}$$

$$L(\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}) = -\log D(\lambda \mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2, \mathbf{y}), \tag{13}$$

forcing the discriminator to treat these mixes as normal data. However, if the two structures differ significantly, it may produce unnatural samples. We compare the performance during the training process with MixUp and SiMix in Appendix F.8. When combined with Pseudo-clean Shuffle, the performance is boosted by a large gap, indicating the model is robust to unseen noise and can even enhance the quality of a well-generated sample. Additionally, Pseudo-clean Shuffle stabilizes the learning process of 20% Asymmetric noise setting, which is often a challenging setting for the noise detector.

Discriminator Training. Training SBDC typically takes 50-200 epochs, while the classifier in TDSM is 4000

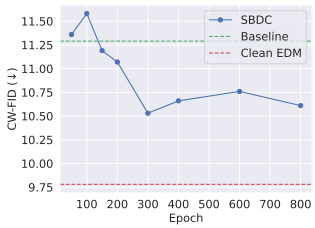


Figure 6. SBDC with different checkpoints of the discriminator.

Wall-clock Time	EDM	SBDC
CIFAR-10	8m35s	8m39s
CLOTHING-1M	23m7s	24m16s

Table 7. Wall-clock inference time (s) per 50k images on EDM (35 steps) with and without SBDC. Guidance is applied on half of the sampling process.

epochs. Figure 6 displays the CW-FID curve during discriminator training. The model quickly outperforms the baseline and stabilizes.

Since our method functions as a plug-and-play module, it can also be integrated with methods like TDSM. We show some generated images from TDSM with/without SBDC in Fig. 7. The experimental results for applying SBDC to TDSM are described in Appendix F.7.

Inference time comparison. All the numbers are measured on 4 NVIDIA A100 GPUs. In Table 7, we report the wall-clock inference time per 100K images on EDM and EDM with SBDC for CIFAR-10 and CLOTHING1M, noting that the inference time of TDSM is the same as EDM. SBDC does not raise any significant delay because the guidance is applied on only half of the diffusion steps.

5. Related Works

5.1. Supervised Learning

In classification tasks, labels are the learning target and directly impact the models' behaviors. Based on learning behaviors, researchers have developed methods for improving models' robustness. Patel and Sastry [37], Zhang et al. [58] showed that the model eventually memorizes noisy data given sufficient training time, however, the model typically remembers clean data before remembering noisy data. The observation leads to techniques such as early stopping [3, 33, 51], curriculum learning [62], ensemble methods [18, 31]. Additionally, noisy models suffer from reduced interpretability, leading to the uncurated representation of data in the latent space. Chen et al. [10], Li et al. [32], Zhu et al. [63] utilized the representations from pre-trained models to mitigate the impact of noise. Cheng et al. [12], Kim et al. [25], Yi et al. [56] introduced regularization techniques to make the learned representation more robust to noise. The generalizability of algorithms for learning with noisy labels is also a question that has been actively studied [15, 19, 57]. In our setting, we aim to learn a model end-to-end without the data pre-processing.

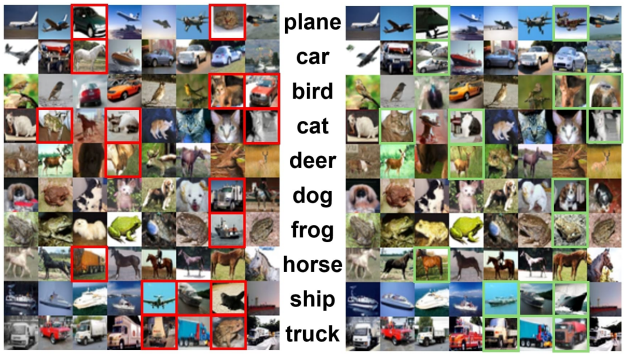


Figure 7. The uncurated generated images of TDSM (left) and TDSM with SBDC (right) on the CIFAR-10 dataset with 50% symmetric noise. We highlight some error images and corrections in red and green squares, respectively.

5.2. Conditional Generative Learning

Several methods modified the GAN architecture [16] to better utilize the label information. Thekumparampil et al. [47] proposes two methods: one adds a calibration objective to reduce mismatch between generated and true labels, and the other boosts confidence in label predictions. Meanwhile, Kaneko et al. [22] integrates a transition probability matrix into the discriminator, enabling implicit inference of the clean label distribution.

Recently, Na et al. [34] integrated transition probability methods into conditional diffusion models, representing a noisy score network as a mixture of clean ones. However, their method omits the instance-independent scenario and approximates the score function using only a fraction of classes to reduce training costs—introducing errors that hinder scalability. To avoid expensive retraining, we leverage the diffusion model's sequential generation to directly correct errors, enabling our approach to handle both instance-dependent and instance-independent cases.

6. Conclusion

This paper tackles noisy labels in conditional generative models by analyzing discriminator signals to differentiate between clean and noisy models. We validate our approach with two novel data augmentation techniques and highlight its potential for further improvement with advanced noise detection. Experiments show our method outperforms others, maintaining efficiency and scalability for large-class datasets. The method can extend to text condition generative modeling with models like CLIP. It would be intriguing to explore zero-shot and few-shot approaches for text noise, especially in privacy and adversarial attacks. We consider this as a future works.

References

[1] Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982. 2

[2] HeeSun Bae, Seungjae Shin, Byeonghu Na, JoonHo Jang, Kyungwoo Song, and Il-Chul Moon. From noisy prediction to true label: Noisy prediction calibration via generative model. In *International Conference on Machine Learning*, pages 1277–1297. PMLR, 2022. 1, 5

[3] Yingbin Bai, Erkun Yang, Bo Han, Yanhua Yang, Jiatong Li, Yinian Mao, Gang Niu, and Tongliang Liu. Understanding and improving early stopping for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34:24392–24403, 2021. 8

[4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1728–1738, 2021. 1

[5] Shane Barratt and Rishi Sharma. A note on the inception score. *arXiv preprint arXiv:1801.01973*, 2018. 5

[6] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020. 1

[7] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VI 13*, pages 446–461. Springer, 2014. 5, 4

[8] Chen-Hao Chao, Wei-Fang Sun, Bo-Wun Cheng, Yi-Chen Lo, Chia-Che Chang, Yu-Lun Liu, Yu-Lin Chang, Chia-Ping Chen, and Chun-Yi Lee. Denoising likelihood score matching for conditional score-based data generation. *ICLR*, 2022. 6

[9] Hao Chen, Yujin Han, Diganta Misra, Xiang Li, Kai Hu, Difan Zou, Masashi Sugiyama, Jindong Wang, and Bhiksha Raj. Slight corruption in pre-training data makes better diffusion models. *Advances in neural information processing systems*, 2024. 2

[10] Jian Chen, Ruiyi Zhang, Tong Yu, Rohan Sharma, Zhiqiang Xu, Tong Sun, and Changyou Chen. Label-retrieval-augmented diffusion models for learning from noisy labels. *Advances in Neural Information Processing Systems*, 36, 2024. 8

[11] Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu. Learning with instance-dependent label noise: A sample sieve approach, 2021. 1, 5, 2, 4

[12] Hao Cheng, Zhaowei Zhu, Xing Sun, and Yang Liu. Mitigating memorization of noisy labels via regularization between representations. *arXiv preprint arXiv:2110.09022*, 2021. 8

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[14] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 4

[15] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020. 8

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2014. 8

[17] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010. 2

[18] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31, 2018. 5, 8, 2

[19] Hrayr Harutyunyan, Kyle Reing, Greg Ver Steeg, and Aram Galstyan. Improving generalization by controlling label-noise information in neural network weights. In *International Conference on Machine Learning*, pages 4071–4081. PMLR, 2020. 8

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 5

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1

[22] Takuhiro Kaneko, Yoshitaka Ushiku, and Tatsuya Harada. Label-noise robust generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2467–2476, 2019. 1, 6, 8

[23] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022. 6

[24] Dongjun Kim, Yeongmin Kim, Se Jung Kwon, Wanmo Kang, and Il-Chul Moon. Refining generative process with discriminator guidance in score-based diffusion models. *ICML*, 2023. 3, 4

[25] Suyeon Kim, Dongha Lee, SeongKu Kang, Sukang Chae, Sanghwan Jang, and Hwanjo Yu. Learning discriminative dynamics with label corruption for noisy label detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22477–22487, 2024. 4, 8

[26] Youngdong Kim, Junho Yim, Juseung Yun, and Junmo Kim. Nlnl: Negative learning for noisy labels. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 101–110, 2019. 5

[27] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 3, 5, 2

[28] Seong Min Kye, Kwanghee Choi, Joonyoung Yi, and Buru Chang. Learning with noisy labels by efficient transition matrix estimation to combat label miscorrection. In *European Conference on Computer Vision*, pages 717–738. Springer, 2022. 1

[29] Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. *arXiv preprint arXiv:2404.07724*, 2024. 4

[30] Yann Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015. 5

[31] Jisoo Lee and Sae-Young Chung. Robust training with ensemble consensus. *arXiv preprint arXiv:1910.09792*, 2019. 8

[32] Jichang Li, Guanbin Li, Feng Liu, and Yizhou Yu. Neighborhood collective estimation for noisy label identification and correction. In *European Conference on Computer Vision*, pages 128–145. Springer, 2022. 8

[33] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020. 8

[34] Byeonghu Na, Yeongmin Kim, HeeSun Bae, Jung Hyun Lee, Se Jung Kwon, Wanmo Kang, and Il chul Moon. Label-noise robust diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2, 5, 6, 8

[35] Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *International Conference on Machine Learning*, pages 7176–7185. PMLR, 2020. 6

[36] Curtis Northcutt, Lu Jiang, and Isaac Chuang. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411, 2021. 4, 5

[37] Deep Patel and PS Sastry. Memorization in deep neural networks: Does the loss function matter? In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 131–142. Springer, 2021. 8

[38] Geoff Pleiss, Tianyi Zhang, Ethan Elenberg, and Kilian Q Weinberger. Identifying mislabeled data using the area under the margin ranking. *Advances in Neural Information Processing Systems*, 33:17044–17056, 2020. 4, 3

[39] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 6

[40] Suman Ravuri and Oriol Vinyals. Classification accuracy score for conditional generative models. *Advances in neural information processing systems*, 32, 2019. 2

[41] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2016. 5

[42] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 1

[43] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 2256–2265, Lille, France, 2015. PMLR. 1

[44] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 4

[45] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 3, 1

[46] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2

[47] Kiran K Thekumparampil, Ashish Khetan, Zinan Lin, and Sewoong Oh. Robustness of conditional gans to noisy labels. *Advances in neural information processing systems*, 31, 2018. 6, 8

[48] David Thiel. Identifying and eliminating csam in generative ml training data and models. Technical report, Technical Report. Stanford University, Palo Alto, CA. https://purl. stanford . . . , 2023. 1

[49] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011. 1

[50] Zijie J Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau. Diffusiondb: A large-scale prompt gallery dataset for text-to-image generative models. *arXiv preprint arXiv:2210.14896*, 2022. 1

[51] Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. Robust early-learning: Hindering the memorization of noisy labels. In *International conference on learning representations*, 2020. 8, 2

[52] Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent label noise: Towards instance-dependent label noise. *Advances in Neural Information Processing Systems*, 33:7597–7610, 2020. 5, 2

[53] Xiaobo Xia, Tongliang Liu, Bo Han, Mingming Gong, Jun Yu, Gang Niu, and Masashi Sugiyama. Sample selection with uncertainty of losses for learning with noisy labels. *arXiv preprint arXiv:2106.00445*, 2021. 5

[54] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2691–2699, 2015. 5, 4

[55] Yu Yao, Tongliang Liu, Bo Han, Mingming Gong, Jiankang Deng, Gang Niu, and Masashi Sugiyama. Dual t: Reduc-

ing estimation error for transition matrix in label-noise learning. *Advances in neural information processing systems*, 33: 7260–7271, 2020. 1

[56] Li Yi, Sheng Liu, Qi She, A Ian McLeod, and Boyu Wang. On learning contrastive representations for learning with noisy labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16682–16691, 2022. 8

[57] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *International conference on machine learning*, pages 7164–7173. PMLR, 2019. 8

[58] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021. 8

[59] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 5

[60] Yivan Zhang, Gang Niu, and Masashi Sugiyama. Learning noise transition matrix from only noisy labels via total variation regularization. In *International Conference on Machine Learning*, pages 12501–12512. PMLR, 2021. 1

[61] Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient gan training. *Advances in neural information processing systems*, 33:7559–7570, 2020. 3

[62] Tianyi Zhou, Shengjie Wang, and Jeff Bilmes. Robust curriculum learning: from clean label detection to noisy label self-correction. In *International Conference on Learning Representations*, 2020. 8

[63] Zhaowei Zhu, Zihao Dong, and Yang Liu. Detecting corrupted labels without training a model to predict. In *International conference on machine learning*, pages 27412–27427. PMLR, 2022. 1, 4, 8